

Informe S8 - Descriptores

Introducción

Hemos calculado diferentes descriptores mediante diferentes métodos:

1. Fourier Descriptors
2. Eccentricity
3. Number of Holes
4. Major Axis Length / Perimeter

Metodología

Binarización

Inicialmente binarizamos de forma simple la imagen mediante un threshold:

```
% Binarizacion
I = imread('Joc_de_caracters.jpg');
I = rgb2gray(I);
I = I < 40;
```

- Imagen Original:

0 1 2 3 4 5 6 7 8 9 B C D F G H J K L M N P R S T V W X Y Z

- Imagen Binarizada:

0 1 2 3 4 5 6 7 8 9 B C D F G H J K L M N P R S T V W X Y Z

Separación

Para poder calcular los descriptores de cada carácter, los separamos mediante el método de las componentes conexas:

```
% Separamos la imagen original entre las diferentes componentes connexas (caracteres).
CC = bwconncomp(I);

% Obtenemos imagenes binarias separadas de los caracteres
caracter = regionprops(CC, 'Image');
```

Cálculo de descriptores

Fourier Descriptors

Para calcular estos descriptores, hemos utilizado una función creada por un usuario de Matlab y publicada en la web de MathWorks. Para mas informacion pinche [aqui](#).

La siguiente función ha sido creada por nosotros simplemente para preparar las imágenes de entrada y pasarsela al método *gfd*.

```
function FD = FourierDescriptor(I)

    BW = I;

    % convertir a escala de grises si no lo esta
    if size(size(BW)) == 3
        BW = rgb2gray(BW);
    end

    % convertir a imagen logica si no lo esta
    if ~islogical(BW)
        BW = BW < 40;
    end

    % centrar objeto en la imagen
    BW = centerobject(BW);

    m = 3;
    n = 3;
    FD = gfd(BW,m,n);

end
```

Para visualizar los datos obtenidos, los hemos introducido en una tabla mediante el siguiente código:

```

% Texto para las tablas
charnames = { '0'; '1'; '2'; '3'; '4'; '5'; '6'; '7'; '8'; ...
              '9'; 'B'; 'C'; 'D'; 'F'; 'G'; 'H'; 'J'; 'K'; 'L'; 'M'; ...
              'N'; 'P'; 'R'; 'S'; 'T'; 'V'; 'W'; 'X'; 'Y'; 'Z'};

% Calculo de los descriptores de Fourier para cada caracter.
FD_0 = FourierDescriptor(caracter(1).Image);
FD_1 = FourierDescriptor(caracter(2).Image);
FD_2 = FourierDescriptor(caracter(3).Image);
FD_3 = FourierDescriptor(caracter(4).Image);
FD_4 = FourierDescriptor(caracter(5).Image);
FD_5 = FourierDescriptor(caracter(6).Image);
FD_6 = FourierDescriptor(caracter(7).Image);
FD_7 = FourierDescriptor(caracter(8).Image);
FD_8 = FourierDescriptor(caracter(9).Image);
FD_9 = FourierDescriptor(caracter(10).Image);
FD_B = FourierDescriptor(caracter(11).Image);
FD_C = FourierDescriptor(caracter(12).Image);
FD_D = FourierDescriptor(caracter(13).Image);
FD_F = FourierDescriptor(caracter(14).Image);
FD_G = FourierDescriptor(caracter(15).Image);
FD_H = FourierDescriptor(caracter(16).Image);
FD_J = FourierDescriptor(caracter(17).Image);
FD_K = FourierDescriptor(caracter(18).Image);
FD_L = FourierDescriptor(caracter(19).Image);
FD_M = FourierDescriptor(caracter(20).Image);
FD_N = FourierDescriptor(caracter(21).Image);
FD_P = FourierDescriptor(caracter(22).Image);
FD_R = FourierDescriptor(caracter(23).Image);
FD_S = FourierDescriptor(caracter(24).Image);
FD_T = FourierDescriptor(caracter(25).Image);
FD_V = FourierDescriptor(caracter(26).Image);
FD_W = FourierDescriptor(caracter(27).Image);
FD_X = FourierDescriptor(caracter(28).Image);
FD_Y = FourierDescriptor(caracter(29).Image);
FD_Z = FourierDescriptor(caracter(30).Image);

fd_names = {'FD_0', 'FD_1', 'FD_2', 'FD_3', 'FD_4', 'FD_5', 'FD_6', ...
            'FD_7', 'FD_8', 'FD_9', 'FD_B', 'FD_C', 'FD_D', 'FD_F', 'FD_G', ...
            'FD_H', 'FD_J', 'FD_K', 'FD_L', 'FD_M', 'FD_N', 'FD_P', 'FD_R', ...
            'FD_S', 'FD_T', 'FD_V', 'FD_W', 'FD_X', 'FD_Y', 'FD_Z'};

T1 = table(FD_0,FD_1,FD_2,FD_3,FD_4,FD_5,FD_6,FD_7,FD_8,FD_9,FD_B, ...
           FD_C,FD_D,FD_F,FD_G,FD_H,FD_J,FD_K,FD_L,FD_M,FD_N, FD_P,FD_R,FD_S, ...
           FD_T,FD_V,FD_W,FD_X,FD_Y,FD_Z, 'VariableNames', fd_names')

```

La tabla con los datos obtenidos es la siguiente:

- Cada fila representa la orden del descriptor.
- Cada columna son los descriptores de un determinado carácter.

	FD_0	FD_1	FD_2	FD_3	FD_4	FD_5	FD_6	FD_7	FD_8	FD_9
1	0.5184	0.2377	0.3528	0.3853	0.2565	0.3753	0.5311	0.1934	0.5304	0.5153
2	0.0215	0.0376	0.0506	0.0676	0.0948	0.0351	0.0621	0.0897	0.0294	0.0216
3	0.0850	0.5540	0.3989	0.2664	0.2353	0.2707	0.1031	0.2559	0.1008	0.1024
4	0.0050	0.0292	0.0737	0.0542	0.1925	0.0614	0.0332	0.2481	0.0200	0.0384
5	0.3851	0.1193	0.2472	0.1983	0.3599	0.1972	0.2340	0.3903	0.2079	0.2193
6	0.0372	0.1141	0.0650	0.1478	0.2143	0.0441	0.1618	0.3160	0.0279	0.1211
7	0.3086	0.2057	0.3753	0.2675	0.0427	0.2232	0.2406	0.0969	0.2973	0.2580
8	0.0512	0.1019	0.1718	0.2323	0.0780	0.0361	0.0207	0.0577	0.0666	0.0592
9	0.0560	0.1087	0.1628	0.1690	0.1386	0.1763	0.1264	0.1657	0.1183	0.1219
10	0.0496	0.0673	0.1555	0.0926	0.1293	0.0137	0.0875	0.1767	0.0219	0.0371
11	0.1821	0.0465	0.1432	0.0989	0.0650	0.1484	0.1047	0.1145	0.0993	0.1190
12	0.0411	0.0391	0.1126	0.1403	0.0645	0.0145	0.1064	0.0552	0.0289	0.0716
13	0.0613	0.0409	0.0567	0.1233	0.0404	0.1124	0.0673	0.0748	0.0674	0.0827
14	0.0178	0.0109	0.0986	0.0564	0.0714	0.0161	0.0440	0.0595	0.0167	0.0301
15	0.0310	0.0436	0.0254	0.0346	0.0130	0.0767	0.0335	0.0513	0.0565	0.0457
16	0.0129	0.0279	0.0946	0.0737	0.0323	0.0044	0.0564	0.0552	0.0162	0.0215

	FD_B	FD_C	FD_D	FD_F	FD_G	FD_H	FD_J	FD_K	FD_L	FD_M
1	0.5040	0.3725	0.4614	0.2754	0.4430	0.4400	0.2221	0.3244	0.2155	0.5512
2	0.0400	0.0984	0.0420	0.0478	0.0179	0.0357	0.0820	0.0702	0.1135	0.0047
3	0.1019	0.2520	0.1339	0.2607	0.1125	0.0630	0.4037	0.1965	0.3809	0.0365
4	0.0571	0.0573	0.0859	0.1054	0.0657	0.0325	0.1881	0.0910	0.1813	0.0563
5	0.2451	0.3642	0.3663	0.2367	0.3275	0.2071	0.1456	0.1717	0.2444	0.2466
6	0.0509	0.2117	0.0824	0.0998	0.0676	0.0221	0.1298	0.1056	0.3341	0.0515
7	0.3021	0.3237	0.3165	0.1390	0.2945	0.2317	0.1367	0.1435	0.2921	0.1309
8	0.0199	0.1086	0.0043	0.1729	0.0165	0.0644	0.1841	0.0931	0.1834	0.0563
9	0.1183	0.2075	0.0772	0.0775	0.0876	0.0462	0.1463	0.0945	0.0059	0.0339
10	0.0513	0.1199	0.0593	0.1073	0.1198	0.0264	0.0881	0.0364	0.0999	0.0356
11	0.1090	0.1172	0.1691	0.0488	0.1753	0.0562	0.1104	0.0342	0.1854	0.0452
12	0.0544	0.1620	0.1042	0.0501	0.0733	0.0185	0.0222	0.0445	0.1823	0.0064
13	0.0959	0.0843	0.0745	0.0721	0.0968	0.0739	0.0147	0.0306	0.0631	0.0517
14	0.0131	0.0320	0.0392	0.0369	0.0302	0.0268	0.0912	0.0291	0.0119	0.0161
15	0.0510	0.0668	0.0168	0.0595	0.0312	0.0305	0.0177	0.0378	0.0295	0.0308
16	0.0400	0.0448	0.0553	0.0269	0.0614	0.0123	0.0402	0.0346	0.0641	0.0063

	FD_N	FD_P	FD_R	FD_S	FD_T	FD_V	FD_W	FD_X	FD_Y	FD_Z
1	0.4925	0.2988	0.3873	0.4048	0.2244	0.3423	0.5843	0.3233	0.2628	0.3577
2	0.0258	0.0413	0.0282	0.0384	0.0717	0.0377	0.0144	0.0149	0.0463	0.0432
3	0.1160	0.1552	0.1258	0.2691	0.4913	0.2143	0.2360	0.3987	0.3342	0.4613
4	0.0219	0.1931	0.0106	0.0708	0.1207	0.2324	0.0703	0.0084	0.2655	0.0560
5	0.2581	0.3223	0.2378	0.2619	0.2184	0.1918	0.1908	0.0836	0.1020	0.3101
6	0.0381	0.1642	0.0621	0.0553	0.1720	0.0783	0.0171	0.0053	0.0630	0.0390
7	0.1503	0.2445	0.2731	0.2249	0.2028	0.2059	0.1526	0.0950	0.1371	0.3445
8	0.0545	0.1488	0.0363	0.0263	0.1160	0.1307	0.0385	0.0019	0.0582	0.0117
9	0.0721	0.0586	0.0907	0.1770	0.1548	0.1449	0.0910	0.0792	0.1035	0.2184
10	0.0399	0.1700	0.1011	0.0417	0.1650	0.0154	0.0113	0.0026	0.0269	0.0382
11	0.0962	0.1661	0.1318	0.1532	0.0436	0.0673	0.0088	0.0780	0.0568	0.2123
12	0.0262	0.0940	0.0644	0.0399	0.0534	0.0498	0.0114	0.0093	0.0553	0.0413
13	0.0555	0.1054	0.0467	0.0916	0.0795	0.0675	0.0130	0.0461	0.0742	0.1045
14	0.0232	0.0867	0.0926	0.0421	0.0513	0.0083	0.0029	0.0022	0.0254	0.0219
15	0.0335	0.0736	0.0323	0.0861	0.0950	0.0250	0.0379	0.0377	0.0101	0.0636
16	0.0163	0.0854	0.0737	0.0249	0.0383	0.0158	0.0158	0.0077	0.0394	0.0355

Eccentricity

Para calcular la excentricidad simplemente utilizamos la función pertinente proporcionada por *regionprops*.

```
% Calculo de los descriptores basados en la excentricidad.  
exc=regionprops(CC,'Eccentricity');  
T2 = table([exc.Eccentricity'],'RowNames', charnames, 'VariableNames', {'Eccentricity'})
```

La tabla obtenida es la siguiente:

	Eccentricity
1 0	0.7443
2 1	0.9822
3 2	0.8952
4 3	0.8780
5 4	0.8624
6 5	0.8862
7 6	0.7733
8 7	0.8706
9 8	0.8015
10 9	0.7916
11 B	0.7909
12 C	0.8522
13 D	0.7868
14 F	0.8914
15 G	0.7903
16 H	0.5757
17 J	0.9312
18 K	0.8013
19 L	0.9246
20 M	0.6331
21 N	0.6243
22 P	0.8278
23 R	0.7929
24 S	0.8651
25 T	0.9357
26 V	0.8548
27 W	0.8484
28 X	0.8785
29 Y	0.8984
30 Z	0.9061

Number of Holes

Para obtener el número de agujeros que tiene cada elemento de la imagen original hemos utilizado la función `EulerNumber` de `regionprops` que nos devuelve un número que representa la cantidad de objetos de la región menos la cantidad de agujeros que se encuentran en los objetos para obtener el valor real de los agujeros hemos restado uno y hemos realizado el valor absoluto.

```
% Calculo del nuemro de agujeros de las imagenes
eulerNum= regionprops(CC,'EulerNumber');
holes= struct2cell(eulerNum);
holes= cell2mat(holes);

for it= 1:size(holes,2)
    holes(1,it)= abs(holes(1,it) - 1);
end

T3 = table([holes]','RowNames', charnames, 'VariableNames', {'Num_Holes'})
```

La tabla obtenida es la siguiente:

	Num_Holes
1 0	1
2 1	0
3 2	0
4 3	0
5 4	0
6 5	0
7 6	1
8 7	0
9 8	2
10 9	1
11 B	2
12 C	0
13 D	1
14 F	0
15 G	0
16 H	0
17 J	0
18 K	0
19 L	0
20 M	0
21 N	0
22 P	1
23 R	1
24 S	0
25 T	0
26 V	0
27 W	0
28 X	0
29 Y	0
30 Z	0

Major Axis Length / Perimeter

Para obtener el valor de la longitud del eje principal de la elipse que contiene el dígito concreto hemos utilizado la función `MajorAxisLength`, y para el perímetro, la función `Perimeter`. Ambas proporcionadas `regionprops`.

```
% Calculo de la longitud del mayor eje respecto al perimetro
majorAxis= regionprops(CC, 'MajoraxisLength');
perimeter= regionprops(CC, 'Perimeter');

MA = cell2mat(struct2cell(majorAxis));
PE = cell2mat(struct2cell(perimeter));

DIV = MA./PE;

T4 = table([DIV]', 'RowNames', charnames, 'VariableNames', {'Major_Axis_Length'})
```

La tabla obtenida es la siguiente:

	MAPerim
1 0	0.4489
2 1	0.5043
3 2	0.3444
4 3	0.3137
5 4	0.3224
6 5	0.3146
7 6	0.3186
8 7	0.3520
9 8	0.4217
10 9	0.3236
11 B	0.4205
12 C	0.3435
13 D	0.4541
14 F	0.3347
15 G	0.2638
16 H	0.2320
17 J	0.4367
18 K	0.2572
19 L	0.4367
20 M	0.2058
21 N	0.2161
22 P	0.3989
23 R	0.3124
24 S	0.3035
25 T	0.4223
26 V	0.2985
27 W	0.2556
28 X	0.2976
29 Y	0.3527
30 Z	0.3586

Resultados

Para diferenciar los caracteres parecidos hemos decidido utilizar estas funciones, en las que mostramos la diferencia máxima que ha habido entre los diferentes descriptores calculados y cuales han sido estos.

Char1	Char2	Descriptor	D(char1)	D(char2)	Delta
0	D	Fourier (4)	0.0050	0.0859	0.0809
8	B	Fourier (4)	0.0200	0.0571	0.0371
M	W	Fourier (3), Eccentricity	0.0365	0.2360	0.1995
			0.6331	0.8484	0.2153
2	S	<u>MajorAxis</u> Perimeter	0.3444	0.3035	0.0409
7	L	<u>MajorAxis</u> Perimeter	0.3520	0.4367	0.0847
L	T	Fourier (3)	0.3809	0.4913	0.1104
		Fourier (6)	0.3341	0.1720	0.1621